

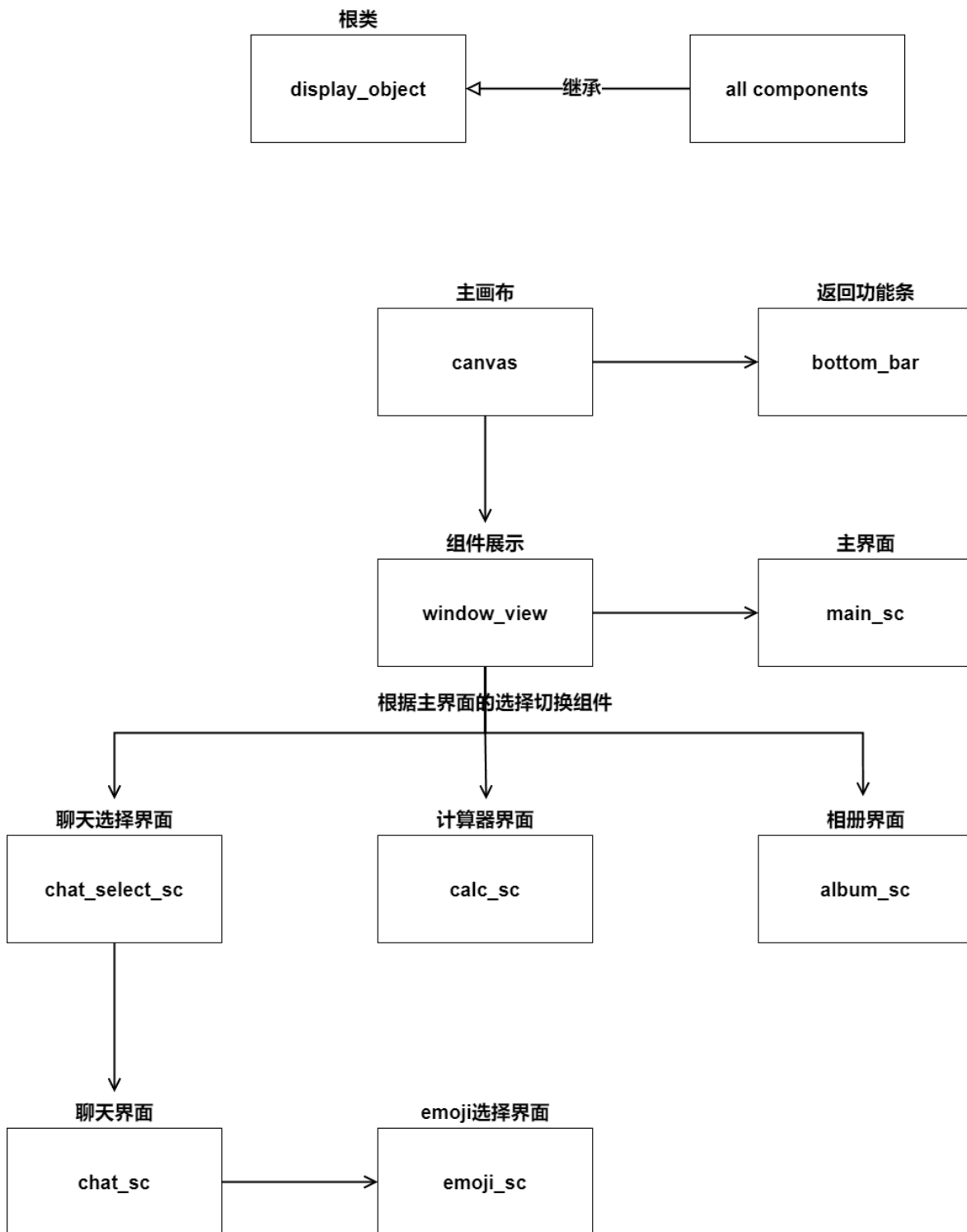
《Embedded System and Microcomputer Principle》 Project Report

Topic	Smartwatch based on STM32					
Group No.	1					
Member 1	SID	12111412	NAME	杨宗奇	Coefficient	1
Member 2	SID	12111515	NAME	刘浩贤	Coefficient	1
Member 3	SID	12110104	NAME	冯泽欣	Coefficient	1
Member 4	SID	12110304	NAME	徐春晖	Coefficient	1
I.System Function (1%)						
<p>我们设计了主页、聊天模块、计算器模块和相册模块，其功能如下。</p> <ol style="list-style-type: none"> 1. 主页： <ol style="list-style-type: none"> (1) 显示自启动以来经过的时间。 (2) 显示该设备的用户名。 (3) 输入不同的组件。 2. 聊天模块 <ol style="list-style-type: none"> (1) 显示其他用户的在线或离线状态，并在状态发生变化时提醒用户。 (2) 与单个用户或所有用户创建聊天。 (3) 通过 uart 串口发送消息。 (4) 提供可以发送的表情符号。 (5) 同时显示消息，可通过用户名进行区分。 (6) 支持断行，翻页和存储历史消息。 3. 计算机模块 <ol style="list-style-type: none"> (1) 基本运算功能和显示 (2) 括号与指数计算功能及显示 (3) 解方程功能及其显示 (4) 二进制运算及显示 (5) 公式过长显示 (6) 基本错误检测 4. 相册模块 <ol style="list-style-type: none"> (1) 图片读取与显示 (2) 相册图片数量显示 (3) 红外遥控器切换图片 						
II.System Design (2%)						
<ol style="list-style-type: none"> 1. 工作原理 <p>我们构建了一个基础的组件框架，所有的组件都继承自一个叫做 display_object (dpo)的根类，该组件有以下几个属性及其作用。</p> <ol style="list-style-type: none"> (1) isVisible，当一个组件的 isVisible 属性为 true 时其拥有功能。 (2) need_render，当一个组件的 need_render 属性为 true 时会自动刷新组件，刷新逻辑为刷新该组件及其子组件，刷新完成后会自动置为 false。 (3) pos，组件相对于父组件的位置。 (4) shape，组件的大小。 (5) click，组件的 click 为 true 时可以表示组件被点击 <p>由于开发板 RAM 和 FLASH 的限制，我们在切换场景时会析构不使用的组件实例，当</p> 						

其要使用时再次创建，将其加入到展示界面 window_view 的子组件中并重新渲染即可使用。按钮点击时间通过判断组件的 click 属性来添加 EVENT，并在主函数的 while 循环中判断事件是否触发执行相应逻辑。

切换场景时我们将 SCENE 置为对应场景，并在下一个循环周期析构当前组件，生成对应新场景实例。最后更新 window view，来达到相对客观的性能。

2. 系统框架图



1. 子模块设计

我们的系统主要划分为三个模块，分别为聊天模块，计算器模块和相册模块。

(1) 聊天模块

点击聊天按钮会进入聊天对象选择界面，在聊天选择界面会显示有哪几个聊天对象以及聊天对象的状态，当且仅当聊天对象在线的时候，才可以发起聊天邀请，当且仅当两个聊天对象都在线的时候，才可以发起群聊邀请。当点击聊天按钮时，系统会自动像被选择的聊天对象发送聊天邀请并进入聊天界面，等候聊天对象进入聊天界面。

对于聊天界面的显示方面，每一页可以容纳 6 条信息/表情包，采用空字符串和图像占位，输入信息后更新对应位置内容的方式显示信息。并使用字符串数组和数字数组存储对应页面的信息和表情，翻页时重新更新内容。

通讯方面，每个用户每一秒钟会广播一条在线信息，格式为“Userx:Online”，当其他用户收到该格式的信息时，更改该用户的状态为在线，并在界面上提示该用户在线，当累计五秒没有检测到在线信息时，更改该用户状态为离线，并在界面上提示该用户离线。对于聊天功能，我们规定发送的信息的格式为“from:to:msg”，当检测到 to 的目的用户是自己，并且自己以及进入了对应的聊天界面的时候，我们才将对应的信息加入到聊天框里面。

(2) 计算器模块

计算器模块，会显示输入框和不同模式下会对应的按钮。并在执行计算后显示输出的结果。

总体：不同的模式会给定不同的按钮，会调用输入框的输出方法来保存按钮的输入字符，所以从软件层面消除了完全非法的字符输入的可能。触发计算（Confirm, = 按钮）后会根据模式的不同调用不同的函数，用来解析输入框目前保存的字符串，并给出一个输出。如果输入合法，给出计算结果；如果输入不合法，显示“ERROR”。

算法：计算算法函数的本质是字符串的输入输出，内部的处理过程都是封装好的，算式 parser 使用了简单的双栈结构，没有实现复杂的 tokenizer。通过对不同符号的优先级比较分析，决定符号压栈或者数字弹栈计算。十进制计算实现了+、-、*、/、(、)、^七个符号的功能，能按照正确的优先级进行计算。二进制计算部分则是先把二进制数字字符串转化拼接为十进制数字字符串，再调用十进制的计算算法。方程计算部分则是对标准形式 ($ax^2+bx+c=0$, ab 不同时为 0) 的算式进行处理，提取出对应的 abc ，再运用一元方程的通项公式进行结果的计算和输出。

显示：算式显示方面，限制算式输入长度为 30，超出后将无法继续输入。dpo 的子框架提供了在光标处输入，光标左右移动，删除光标处字符的功能。计算器模块扩展了超长输入：在输入长于最大显示时，只显示部分算式，通过左右移动光标可以查看完整的算式内容。

(3) 相册模块

相册模块使用 jpg 解析函数和正点原子的 piclib 处理图像读取问题。

SD 卡移植：

本方案采用正点原子 SD 卡驱动方案，mmcsd 模块和 FATFS 外部组件。最初移植时发现了 FLASH 爆满的问题，经检查后发现其包含了巨大的 GBK 转译数组。经查阅资料后，发现可以使用外部 Flash 进行字库和转译数组的存储，因此使用 W25QXX 存放了该数组以及 12 16 24 大小的 GKB 编码中文字模（虽然任务并未要求，但我们实际支持中文显示）

方案：

场景初始化时遍历 PICTURE/文件夹下所有文件，添加文件名至索引列表，以供后续遍历。初始化 image_index 为 0，显示第一张图片（如果没有图片则显示 None）。为了更方便的实现图片显示，我们在 image 组件下加入了索引选项。如果存在 SD 卡索引目录则加载 SD

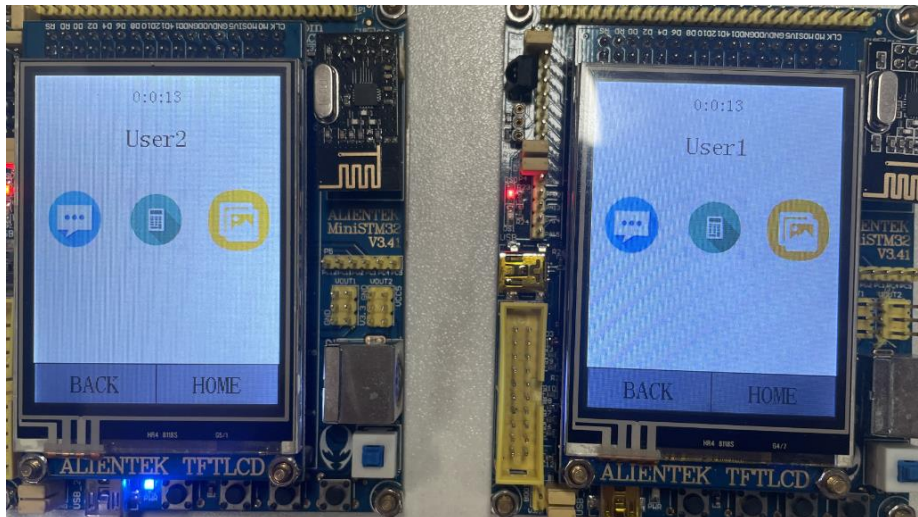
卡中的图片，若没有则加载 const char 图片数组。

遥控：

使用红外遥控 remote 模块，周期扫描当前按下的按键并更新 EVENT，识别对应 EVENT（如 left right）更新 image_index（+1、-1）并更新场景。

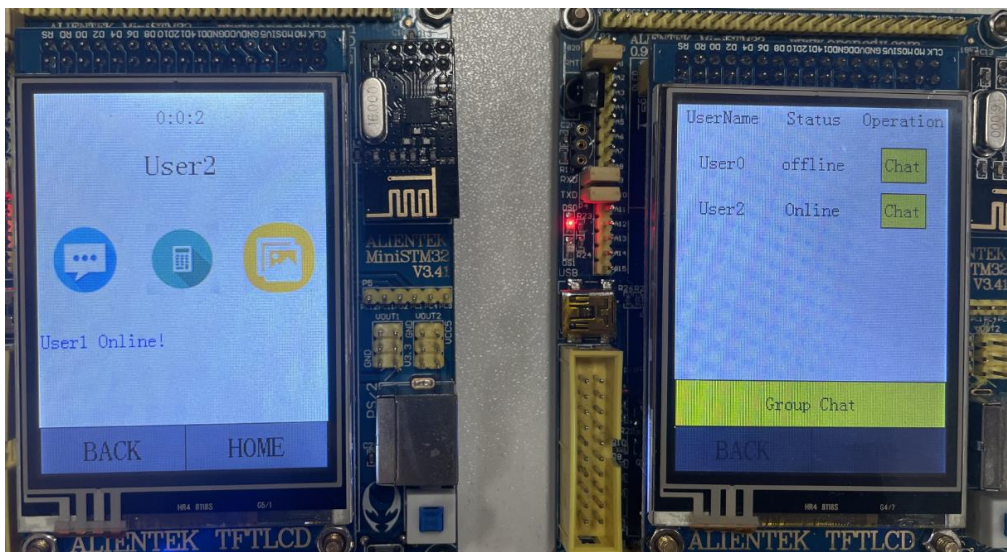
III.Results (screenshots and hardware photos) (1%)

(1) 主界面

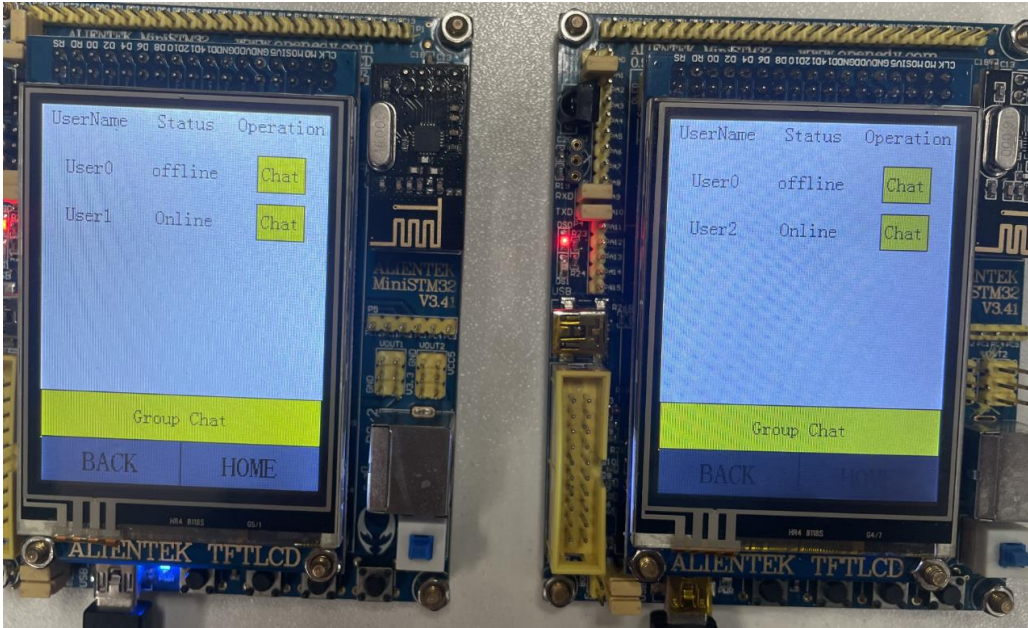


(2) 聊天模块

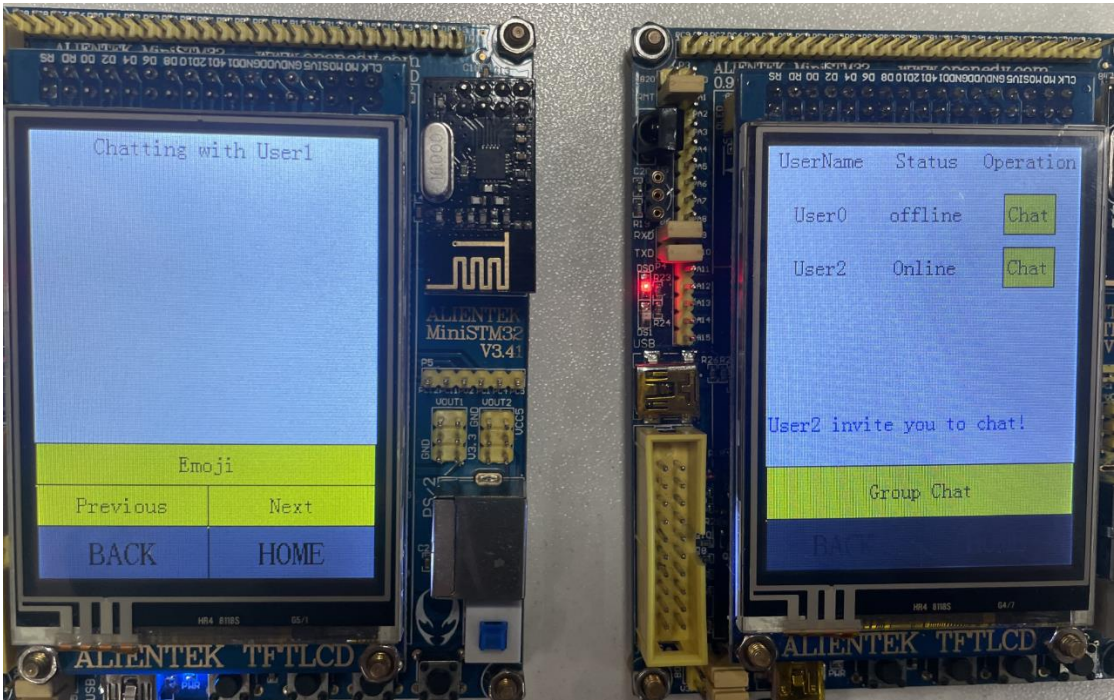
1、上线检测



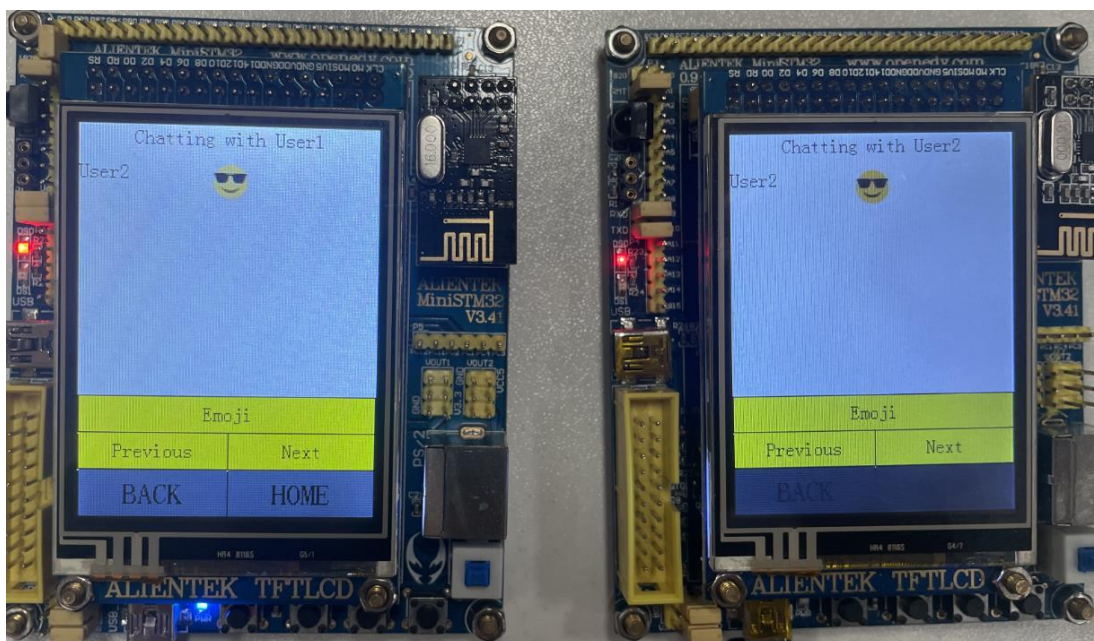
2、上线/下线状态改变



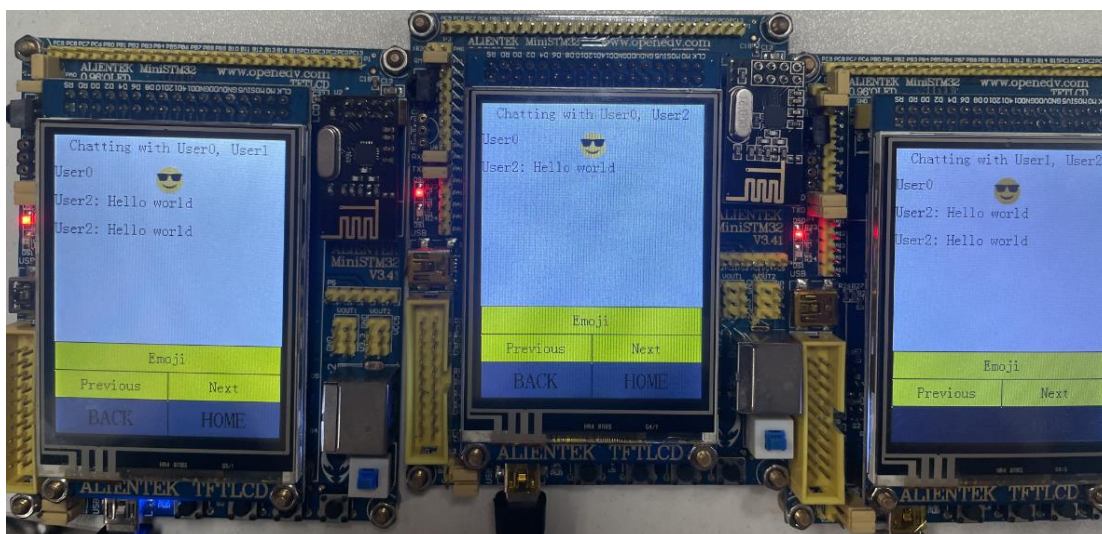
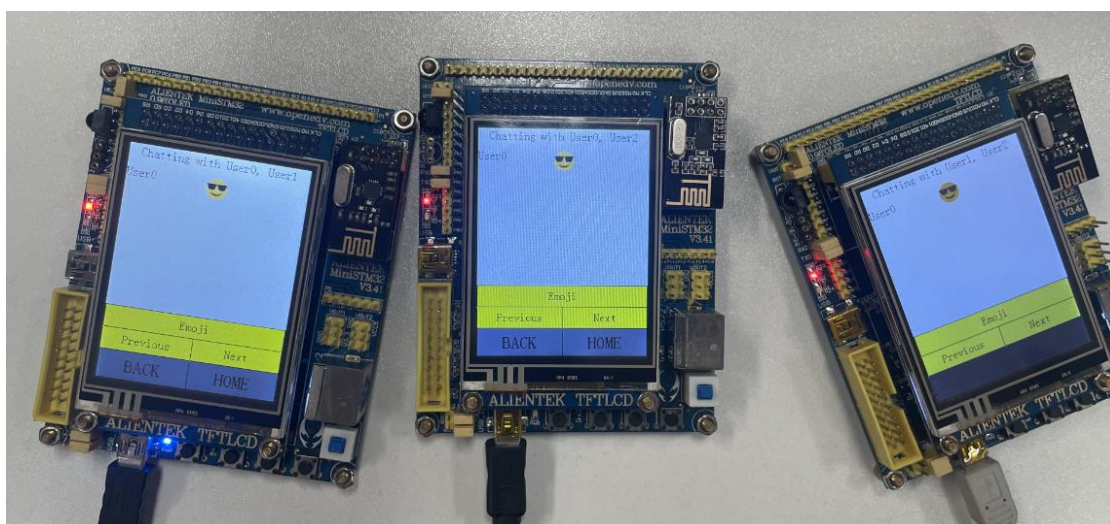
3、聊天邀请



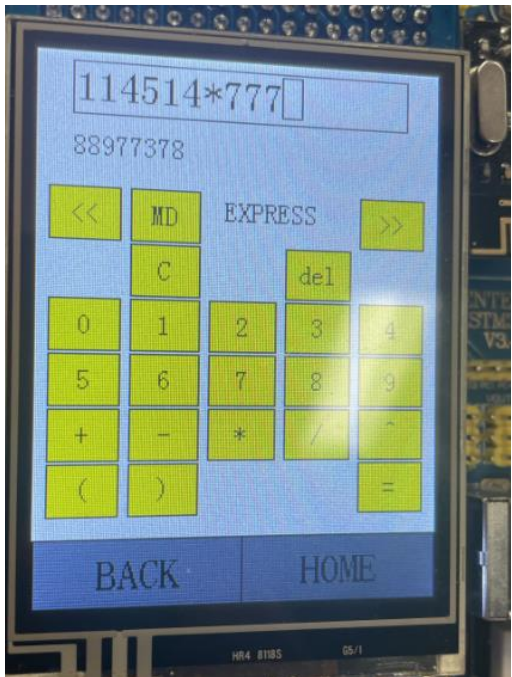
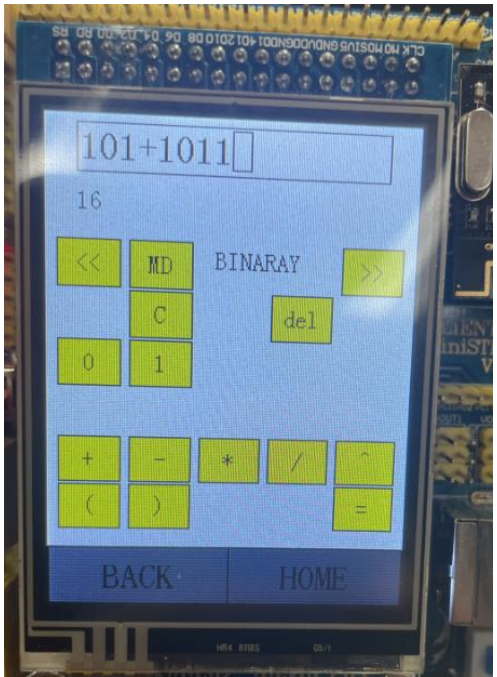
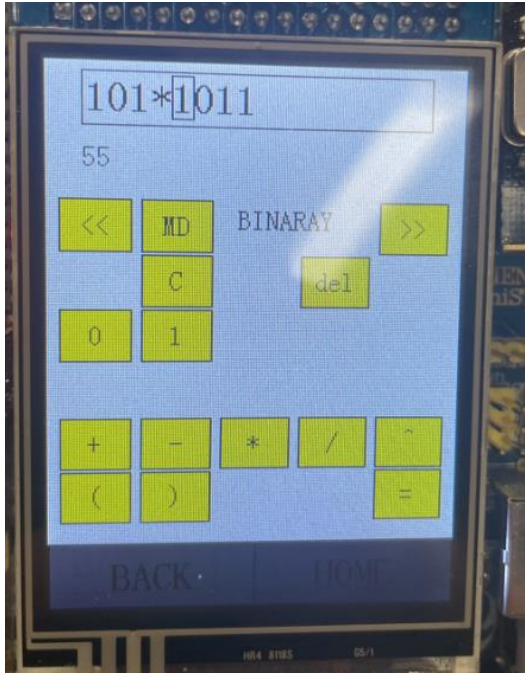
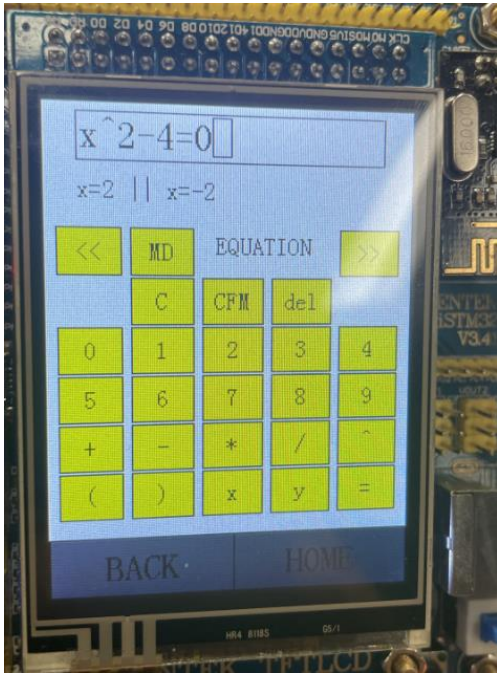
4、单人聊天



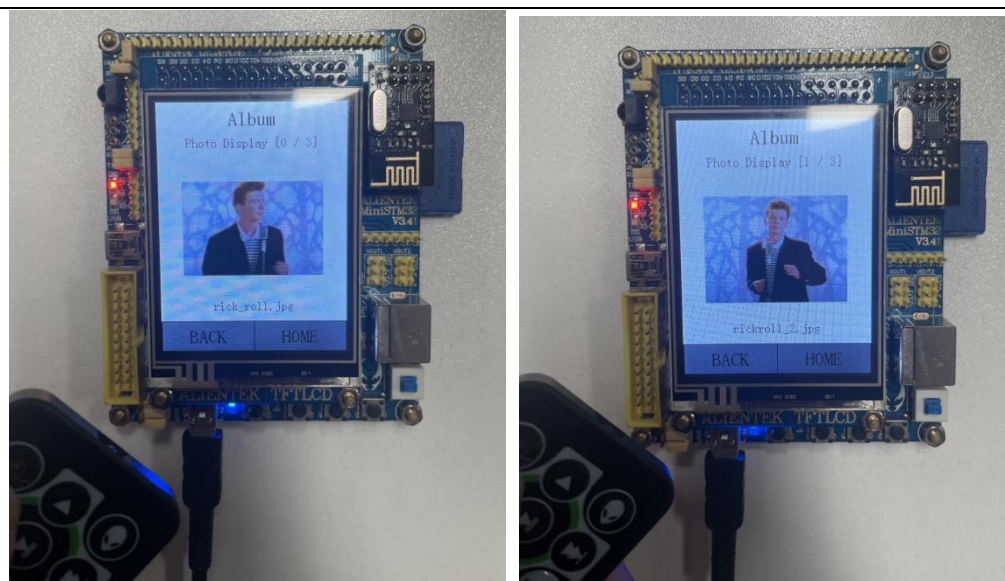
5、多人聊天



(3) 计算器功能（解方程，二进制计算，十进制计算）



(4) 相册模块



IV. Work allocation description (1%)

- (1) Member1: 冯泽欣, 基本框架搭建和相册模块的实现
- (2) Member2: 徐春晖, 计算器模块的实现。
- (3) Member3: 杨宗奇, 主页面和聊天模块接口的实现。
- (4) Member4: 刘浩贤, 用户状态监测和聊天通讯功能的实现。

V. Problems encountered and solutions (1%)

问题 1: 多个组件同时存在会出现 RAM 和 FLASH 容量溢出的问题

解决方案: 在组件不使用时进行析构来释放内存, 再次使用时重新生成组件实例, 对于需要存储信息的组件, 写一个析构函数来保存信息。并且将字库烧录到外置 Flash W25Qxx 中

问题 2: 在聊天通讯时频繁出现丢包或者碰撞的问题

解决方案: 降低发包频率, 并尽量不同时进行通讯操作, 降低出现丢包或者碰撞问题的可能性。

问题 3: 在计算器 IO 部分, 尝试直接格式化 / std::to_string() 输出 double 类型的计算结果会导致软件崩溃错误

解决方案: 首先研究发现可以打开 STM32 开发板浮点数 IO 的功能, 为了不引入额外的软件开销, 写了一个简单的字符串输出算法将接受的 double 类型的数据通过逐位取整逼近 int 的方式格式化拼接成字符串, 然后输出。

VI. Personal proposal (4%)

基础

1. 制作一个音乐播放器
2. 音乐存在 SD 卡目录下 (本次 Project 实现过)
3. 拥有 LCD 进度条, 暂停, 切换歌曲功能 (LCD 实现过)
4. 使用蜂鸣器播放音乐 (需要预先处理一段音乐) 可行性参考

[STM32F103 的 DAC——实现音频输出_dac 生成音频-CSDN 博客](#)

[基于 STM32F103, 用蜂鸣器播放歌曲_stm32f103 蜂鸣器音乐-CSDN 博客](#)

5. 设备间通过 2.4Ghz 模块或者蓝牙模块在设备间分享音乐，（实现过 2.4Ghz 通讯，半双工即可，需要设计一套 request ack 类似于计网的可信传输协议）
6. 可以通过 USB 接口从电脑向 SD 卡下载音乐，参考
[【正点原子 STM32 连载】第六十章 USB 读卡器实验摘自【正点原子】STM32F103 战舰开发指南 V1.2_stm32f103 usb bulk-CSDN 博客](#)

Bonus:

1. 使用蓝牙模块（王老师有提到学校有就不再提供额外链接）与手机进行分享
2. 支持歌曲专辑封面显示（同名 jpg 文件即可，本次实现过）
3. 更好的音质（采样、播放手段）（开放性）
4. 标准化格式音乐支持（mp3, wav, 较为困难）
5. 2.4Ghz 多设备同步播放、多声道环绕音（使用 2.4G、蓝牙同步播放进度，周期同步）